4

Using the Flow of Story for Automatic Video Skimming

Songhao Zhu[1] and Yuncai Liu[2]
[1]*play_tree@163.com*, [2]*whomliu@sjtu.edu.cn*
Institute of Image Processing and Pattern Recognition,
Shanghai Jiao tong University
800, Don chuan Road, Shanghai 200240, China

Abstract: In this chapter, we present a novel scheme to incorporate the flow of story to select salient scenarios for generating semantically meaningful skimming of continuously recorded video such as a movie. We obtain clues of a movie from scenario editing rules and movie production techniques, which are commonly adopted in the process of video making to express the flow of the movie explicitly. The generated skimming not only provides a bird view of the original video but also help viewers understand the overall story. Furthermore, different types of video skimming can be appropriately generated with respect to different user requirements. Experimental results show that the proposed scheme is a feasible solution for the management of video repository and online review services.

1. Introduction

Nowadays the huge amount of video material stored in multimedia repositories makes the browsing, retrieval and delivery of such content a very slow and usually difficult task. According to a report from the China Central Television Web site, the total click count for movie review services reaches almost 100 million per day. In other words, movie occupies up to 50% compared to other genres of review services. This is largely due to the fact that it will take a viewer at least several minutes to watch a film episode before he can understand what happens and why it happens in the episode. In such instance, it is desirable to quickly browse video content in limited bandwidth, which can be achieved by automatic video content skimming.

Video skimming is defined as a sequence of moving images that compactly present the content of a video without losing the main information [1]. According to different characteristics of video content, we can categorize videos into two genres: event-oriented videos and story-oriented videos. An event-oriented video, such as sports or news, has a predetermined structure, and viewers can obtain entertainment or information even by only watching a few interesting parts. However, a story-oriented video, e.g., a movie, has no explicit structure, and a viewer needs to watch the whole video if he wants to know the story. Therefore, the skimming of an event-based video aims to compactly provide entertainment or information to viewers, whereas that of a story-oriented video aims to render the impression of the con-

tent of entire video in a short period of time. Several techniques have been proposed to generate skimming for event-oriented videos by exploiting the well-known structures of videos. In [2-4], a highlight of a news video can be a collection of anchor shots, while that of a soccer video can be generated as a collection of goal shots [5-6]. However, issues related to the production of an abstraction of a story-oriented video have not yet been widely studied.

One of the major challenges in generating skimming for a story-oriented video is that the cognitive process of viewers in perceiving the progress of a story while watching a video is not well understood. Most literatures [7-13] focus on the skimming production which presents the overall mood of the video, rather than help in perceiving the progress of the story. Sundaram *et al.* [14] first integrate viewers' comprehension of video contents into the generation of video skimming. More specifically, the video skimming is constructed by reducing scene duration, i.e., removing redundant shots and frames in scenes while preserving salient shots. To guide the reduction of scenes, a shot-utility function is utilized to model the degree of human understanding of a video shot with respect to its visual complexity and duration.

We propose a framework based on the progress of a story to perform the skimming of movie. As aforementioned, [14] solves the issue of video skimming at the scene level, while the proposed approach attempts to comprehend video contents from the progress and viewer's understanding of the overall story, which is realized as a sequence of scenarios and their relationships [15]. Considering that a scenario is usually captured by a scene, our approach can be considered as a higher level approach and a complement to Sundaram's scene-based scheme.

For the proposed framework based on the story progress, we first segment a video into shots based on the property of two-dimension histogram entropy of image pixels. Next, we generate semantically meaning scenarios by exploiting the spatio-temporal correlation among detected shots. Finally, we exploit general rules of special scenario and common techniques of movie production to grasp the flow of a story according to the degree of progress between scenarios to the overall story, which is evaluated in terms of the intensity of the interrelationships between scenarios.

To demonstrate the effectiveness of the proposed framework, we develop a video skimming system to conduct a set of comparison experiments using objective evaluation criteria, such as precision and recall. Furthermore, we also perform subjective tests to see the viewer's story understanding as well as to see the viewer's satisfaction. Results consistently show the advantages of the proposed framework. The basic idea of the proposed framework is shown in Figure 1.

The main contributions of the proposed framework are as follows.

- We propose a shot boundary detection approach by utilizing the property of two-dimension histogram entropy of image pixels;
- We generate scenarios of a video based on the spatio-temporal correlation between detected shots;
- We exploit clues about the progress of a story to implement the generation of video skimming.

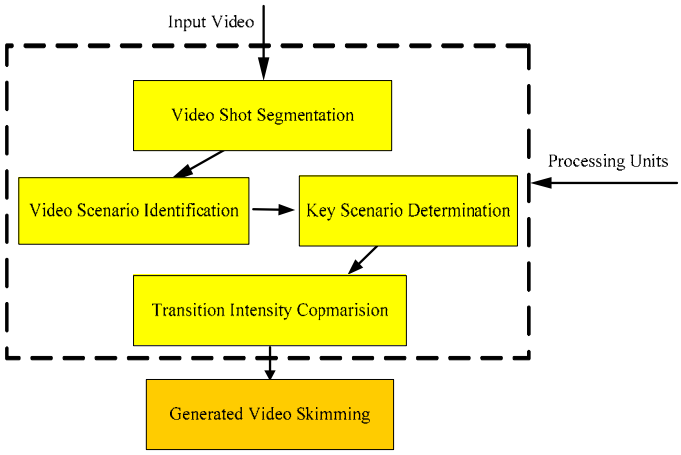


Fig. 1. Overview of the proposed framework.

The remainder of this chapter is organized as follows. In Sections 2 and 3, we introduce the process of temporal video segmentation and scenario detection, respectively. General rules of special scenario and common techniques of movie making are described in Section 4. Section 5 presents the story-oriented video skimming approach. Experimental results are provided in Section 6, followed by concluding remarks in Section 7.

2. Temporal Video Segmentation

In this section, we first depict the property of two-dimension entropy of image pixels, and then employ a progressive algorithm to detect shot boundaries.

2.1. Property of Two-Dimension Histogram Entropy

According to [16], two-dimensional histogram entropy of pixels can be used to describe the distribution of information of a grey-level image. Since the horizontal or vertical change of the gray value of pixels can not well represent the total characteristics of an image, we improve the construction of components of the two-dimensional histogram. More specifically, the gray value of pixel x and the average gray value of pixels in its 3×3 neighborhoods are replaced by the average gray value of pixels in its four neighborhoods a^{x}_{1} and average gray value of other four pixels located in the corner of its 3×3 neighborhoods a^{x}_{2}, respectively. The formulation of the two-dimensional histogram entropy of an image I is:

$$\begin{cases} E = \sum_{i=0}^{L-1} \sum_{j=0}^{L-1} E_{ij}, \; E_{ij} = -N_{ij} * \ln N_{ij} \\ N_{ij} = H_{ij} / H, \qquad H = \sum_{i=0}^{L-1} \sum_{j=0}^{L-1} H_{ij} \end{cases} \tag{1}$$

where H_{ij} presents the two-dimensional histogram of the two-dimensional channel (i, j), and L is set at 256 for gray pixel. Here, i and j represents $a^x{}_1$ and $a^x{}_2$, respectively.
Figure 2 lists some examples to show the relationship between the information contained in image and corresponding value of two-dimensional histogram entropy.
Based on our daily observation and movie editorial techniques, the information contained in images will usually change when there exists shot transition. In such instance, the issue of the detection of shot boundaries can be resolved by using the property of two-dimensional histogram entropy.

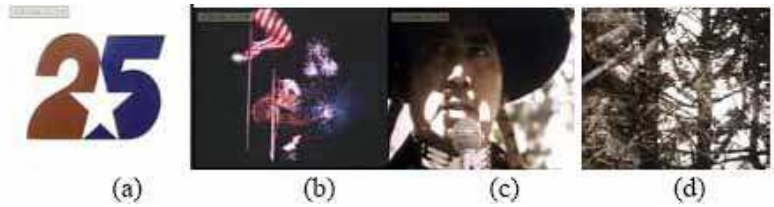


Fig. 2. Examples of image with more information having bigger two-dimension histogram entropy value.

2.2. Shot Boundary Detection

Our shot boundary detection algorithm consists of two levels: one is the coarse identification of candidate segmentation boundaries and the other is the refinement processing of candidate segmentation boundaries. One of the advantages of this progressive scheme is that it can speed up the overall processing with reduced number of video frames and effectively detect the gradual transitions (such as fade in/out, dissolve and wipe).

2.2.1. Coarse Boundary Identification

To coarsely identify candidate boundaries, we first perform temporal subsampling of the input video. In this chapter, the sub-sample rate is typically set to be 10 frames per second. Then, image space of each frame f in the temporally sub-sampled sequence is divided into five regions as shown in Figure 3 due to the following fact. In most cases, the difference between images can be determined according to the information of region ①.

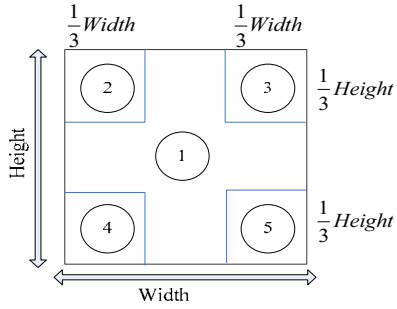


Fig. 3. Selected region for coarse candidate boundaries.

We now perform the coarse identification for candidate boundaries on the extracted main-image sequences in temporally sub-sampled sequence. Given the sequence of two-dimensional entropy difference between successive main-images, a set of candidate boundaries can be detected by appropriate thresholding. Instead of the global thresholding scheme in [17], we make use of the local thresholding approach presented by [18].

2.2.2. Exact location Determination

For those coarse candidate boundaries from the sub-sampled sequence, it is necessary to identify their exact locations at original temporal resolution. If the temporal sub-sampling rate is *N* (here *N=6*) and the frame number of candidate boundary is *m*, the precise position of shot transition will then fall into the localized neighbor region *r* centered on the candidate boundary *m* which contains frames from number *m*N-2*N* to *m*N+2*N*. That is, the exact position of shot transition *k* is achieved by picking a minimum valley within the neighbor region. The overall coarse-to-fine refinement procedure under a certain sub-sample rate is shown in Figure 4.

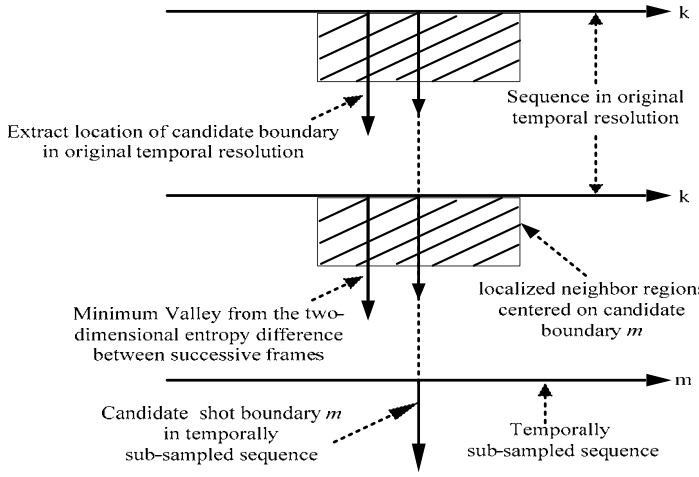


Fig. 4. Overview of the coarse-to-fine refinement procedure.

2.3. Shot Transition Verification

Since coarse candidate boundaries are detected using increased inter-frame difference in sub-sampling sequence, false detections maybe generated in the set of candidate boundary. Therefore, it is indispensable to perform verification of each identified shot transition frame *k*.

To robustly verify detected shot boundaries with respect to various transitions and special effects, we present a novel method to evaluate the similarity of visual content of frames within a local neighbor region *r* as shown in Figure 5 (a). More specifically, suppose the size of *r* is *2*N+1*. Then, feature set of left and right neighbor region, F_L and F_R, are:

$$\begin{cases} F_L = \left[F_{k-2*N}, \ldots\ldots, F_{k-1} \right] \\ F_R = \left[F_{k+1}, \ldots\ldots, F_{k+2*N} \right] \end{cases} \tag{2}$$

where F_i is a 5-dimension vector of 2-dimension entropy extracted from five regions as shown in Figure 3.

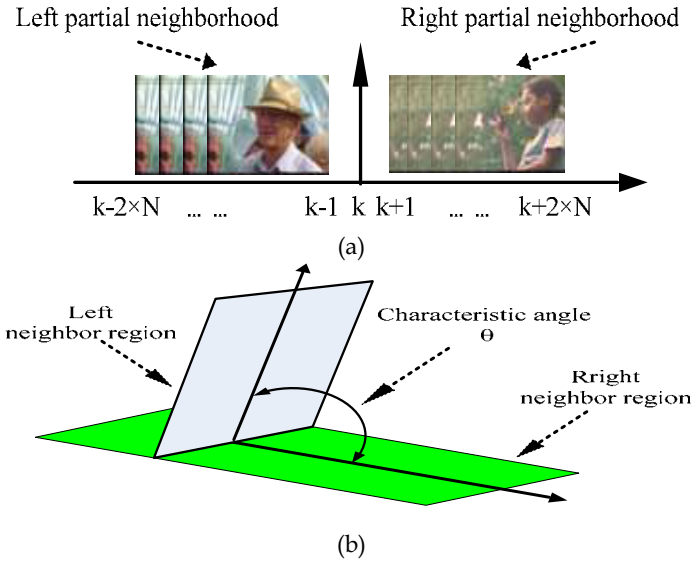
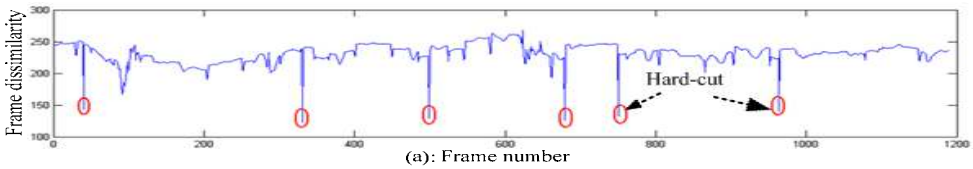


Fig. 5. Illustration of (a): local neighbor region centered on transition frame k and (b): Characteristic angle.

Principal component analysis is then utilized to preserve sub-images with more information for later computation of characteristic angle. P_L and P_R are corresponding orthonormal matrices of F_L and F_R by preserving n eigenvectors corresponding to first n largest eigenvalues. Next, singular value decomposition is performed on $P_L{}^T P_R$ to obtain the characteristic angles θ as in Figure 5 (b):

$$\begin{cases} \theta = \arccos(\delta) = \arccos\left[\max\left(\delta_1, \delta_2, ..., \delta_n\right)\right] \\ U^{-1}\left(P_L^T P_R\right)\left(V^T\right)^{-1} = \Sigma = Diag\left(\delta_1, \delta_2, ..., \delta_n\right) \end{cases} \tag{3}$$

Finally, any detected shot transition frame is considered as actual one only if its characteristic angles θ is bigger than a predefined threshold. Figure 6 shows two examples of shot detection results.



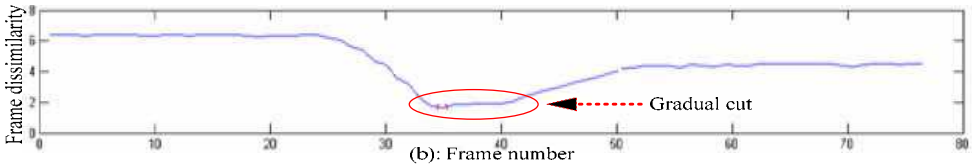


Fig. 6. Shot detection results for (a): abrupt transition and (b): gradual transition.

2.4. Key Frames Extraction within Shots

Representing the content of a video shot concisely is a necessary step for various video processing. In this chapter, a two-pass algorithm is used to complete the task of the selection of key-frames. Given a shot $Sh=\{f_1, f_2, \ldots\ldots, f_U\}$ with U frames, the process of key-frames choosing is then described as follows.

- First, f_1 is chosen as the first key-frame into the:

$$KFS = \{Kf_1\} = \{Kf_N\} = \{f_1\},\ N = 1 \tag{4}$$

where N is the total number of key-frame.

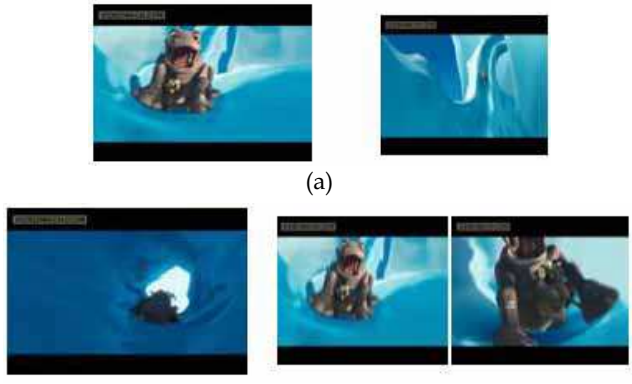
- For each frame f_m behind the current key-frame Kf_N, the value of two-dimension histogram intersection between f_m and each key-frame from $KFS=\{Kf_1, Kf_2,\ldots\ldots, Kf_N\}$ is then computed:

$$\begin{cases} FraSim(f_m, Kf_n) = \sum_{i=0}^{L-1}\sum_{j=0}^{L-1}\min(H_{ij}^{f_m}, H_{ij}^{Kf_n}) \\ FraSim(f_m, KFS) = \{FraSim(f_m, Kf_n)\},\ 1 \le n \le N \end{cases} \tag{5}$$

If values in $FraSim(f_m, KFS)$ are all greater than a pre-fixed threshold decided using the method in [17], then f_m is absorbed into the key-frame set KFS as a new key-frame:

$$KFS = \{Kf_1, Kf_2, \ldots\ldots, Kf_N, f_m\} = \{Kf_1, Kf_2, \ldots\ldots, Kf_N, Kf_{N+1}\} \tag{6}$$

Figure 7 shows some examples of key frame extraction from different shots, where the most left image in each row is the detected shot followed by its key frame(s).



(a)

(b)

(c)

Fig. 7. Examples of key frame extraction with respect to visual content.

3. Scenario Boundary Detection

In this section, using the detected shots we construct a spatial correlation chain within certain temporal constraint, and then determine scenario boundary with the constructed spatio-temporal correlation chain.

3.1. Temporal Constraint Analysis

To avoid under-segmentation or over-segmentation of scenarios, we introduce the concept of temporal constraint similar to [20]. However, compared to [20], both the form and goal are different. Specifically, if the temporal spam between shots exceeds certain constraint, then they will not be grouped into the same scenario although their visual content is similar. For example, two shots in Figure 8 (a) are from different scenarios of movie 'X Man III' although there exists certain similarity between these two shots in the visual aspect. Two shots in Figure 8 (b) are another group of comparison shots, where the image information in terms of human understanding is also fully different.

(a) (b)

Fig. 8. Illustration the importance of temporal constraint. (a) and (b) are two groups of comparison shots both from different scenarios though they share certain visual information.

Temporal constraint (analysis window) depicts the largest number of shots contained in a scenario, which means only shots falling into the analysis window and satisfying spatial correlation can then be grouped into the same scenario.

The choice of the size of analysis window needs to be seriously considered due to its great impact on final video skimming. On the one hand, if this value is too large, shots from different scenarios may be clustered into the same one. That is, too large size will bring about the under-segmentation of scenario. On the other hand, if this value is too small, shots belonging to the same scenario may have different scenario labels. In this case, the issue of the over-segmentation of scenario will occur. Therefore, the size of analysis window is set to be fifty as default based on the heuristic rule. Final experimental results demonstrate that the best accuracy of scenario segmentation is obtained with this default value compared with other values.

3.2. Scenario boundary Identification

To accurately segment scenario, besides the information of visual aspect, the factor of temporal aspect is also taken into account. Different scenarios are obtained by the following passes.

(1) The similarity between two key frames p and q from different shots is measured by the Euclidean distance:

$$S_f(p,q) = \sqrt{\sum_{i \in allbins} \left(\left| F_p^i \right| - \left| F_q^i \right| \right)^2} \tag{7}$$

where F_p^i denotes the i^{th} two dimension entropy of frames p.

(2) The measurement of the dissimilarity between shot m with M key frames and shot n with N key frames is the average distance across all the possible pairs of key frames in each shot:

$$S_{shot}^{m,n} = \frac{\sum_{i \in M} \sum_{j \in N} S_f(i,j)}{M * N} \tag{8}$$

(3) Within one given analysis window Tw, the shot similarities between all pairs of shots are computed by:

$$S_{scenario}^{T_w} = S_{shot}^{m,n}, \ m \in T_w, \ n \in T_w \tag{9}$$

(4) For the current shot c within Tw, all the subsequent shots sharing similar visual content ($f_1, f_2 \ldots f_{s-1}$ and f_s) are:

$$S_{shot}^{c,k} < T_{scenario}, \ c < k \le T_w \tag{10}$$

where $T_{scenario}$ is the threshold. Furthermore, shot f_s is chosen as the current shot c for next processing.

(5) For shots between f_{s-1} and f_s, similar shots ($r_1, r_2 \ldots r_{t-1}$ and r_t) between shot f_s and T_w are located by:

$$S_{shot}^{c,k} < T_{scenario}, f_{s-1} < c < f_s, f_s < k \le T_w \tag{11}$$

Furthermore, shot r_t is also chosen as the current shot c for further processing.

(6) Repeat step 4 or / and 5 until the end of T_w, and there exists two types of scenario boundary. One is if the iterative process stops at step 4, then shot f_s is the current scenario boundary and shot f_s + 1 is the beginning shot for the next scenario. The other is if the iterative process stops at step 5, then shot r_t is the current scenario boundary and shot r_t +1 is the beginning shot for the next scenario.

Figure 9 shows an example of the scenario boundary detection procedure for one testing video: interview video. Shot f_0 is first chosen as the current shot c, and shots f_1, f_2, f_3, and f_4 are visually similar shots according to inequality (7). Next, shot f_4 is selected as the current shot c, and shots r_1, r_2 and r_3 are visually similar shots satisfying inequality (8). Then, shot r_3 becomes the current shot c with respect to the scheme of scenario identification. Since there are no more shots meeting the conditions of inequality (8), shot r_3 is the last shot for current scenario X, and the procedure of the boundary detection for scenario X ends. That is, scenario X is composed of all of the shots from shot f_0 to shot r_3.

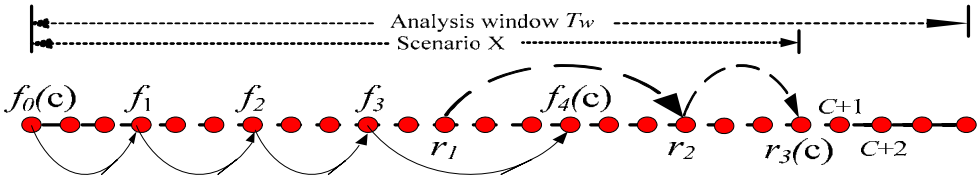


Fig. 9. Procedure of spatio-temporal coherence clustering for current scenario X. Red circles denotes shots within one given analysis Tw. Shots (f_0, f_1, f_2, f_3, f_4, and f_s) linked by solid line are similar ones generated by inequality (8). Shots (r_1, r_2, rand r_3) linked by dotted line are similar ones satisfying inequality (9). Scenario X is composed of all of the shots from shot f_0 to shot r_3.

3.3. Key Frames Extraction within Scenarios

Just like the purpose of extracting key frames from a video shot, the main content of a video scenario can also be concisely represented by corresponding key frames. Key frames of a scenario are chosen from the key frames of each shot within the scenario using the same method as discussed in subsection 2.4.

4. Scenario editing rules and movie making techniques

To construct a skimming for story-oriented videos (such as movies), it is necessary for viewers to grasp the flow of the story from scenario editing rules. According to [21], these rules can effectively describe the semantics of a scenario and so are commonly utilized in the movie production process. Moreover, story progress can be obtained from movie making techniques that are frequently adopted in movie making procedure to articulate a story. Next, we will explore the form of movie from these rules and techniques to introduce common types of key scenario and metrics of the story progress.

Based on our observation and the scenario editing rules, there exist several types of scenarios named key scenarios. These key scenarios often present vital information and hence always attract viewers' attention:

1) Dialog scenario: see Figure 10 (a); 2) Suspense scenario: see (b); 3) Action scenario: see (c).

(a) Dialog scenario

(b) Suspense scenario

(c) Action scenario

Fig. 10. Examples of different types of scenarios.

In [22], a film consisting of scenarios is regarded as a system and the flow of a film is depicted as the interrelationships between key scenarios. Generally, the interrelationship between scenarios is often captured by different types of transitions including temporal transition, spatial transition and rhythmic transition. From the point of view of film editing techniques and human understanding, the more a story progress between scenarios, the more the transition intensity is. Therefore, the intensity of scenario transition is considered as an important metric for evaluating the flow of a film as well as an important basic for the production of video skimming.

5. Video Skimming Approach

This section details the proposed story-oriented video skimming approach, which aims to enable viewers to grasp main content of original video by watching the generated short skimming. Thus, final video skimming should comprise those pairs of video segments with more contributes to the flow of a story. At the same time, the total duration of selected segments should satisfy the user-defined target duration.

The structure of this section is organized as follows. Related symbols are first introduced. Next, key scenarios are identified using visual and audio features. Then, an Intensity of Flow (IoF) function is defined to depict the transition intensity between key scenarios to the whole story. Finally, a Story Structure Tree (SST) is adopted to describe the interrelationship between scenarios in a story-oriented video, and the proposed skimming approach in subsequent sections is described with the SST.

5.1. Related Symbols

Table 1 lists the symbols used in the proposed story-oriented video skimming approach.

Symbol	Meaning
V	A story-oriented video
Sh_i	*i-th* shot in V
Ks_j	*j-th* key scenario in V
$ShKs(Sh_i,Ks_j)$	*i-th* shot of *j-th* key scenario in V
$N(Sh_i,Ks_j,Kf_s)$	Number of characters of *s-th* key frame of *i-th* shot from *j-th* key scenario in V
$TI(Ks_i, Ks_j)$	Transition intensity between key scenario *i* and *j*

Table 1. Related symbols.

5.2. Key Scenarios Identification

According to the discussion is Section 4, a story-oriented video is composed of a sequence of scenarios including key scenarios as shown in Figure 10 and remaining scenarios. Recall

that our purpose in this chapter is to achieve a highly compact skimming of a long video. Therefore, finally generated skimming contains only key scenarios while omitting remaining scenarios. Next, we will detail the process of key scenarios identification.
Based on our observation and scenario editing rules, typical characteristics of three kinds of key scenarios discussed in Section 4 are described as below.

- Dialogue scenario: faces with similar spatial position and similar size, a sequence of shots with low activity intensity, and strong similarity between shots.
- Action scenario: a set of shots with short duration, and intensive activity or audio energy.
- Suspense scenario: a set of shots with low average intensity distribution, a long period of low audio energy, and low activity intensity followed by a sudden change either in sound track or in activity intensity or both.

Next, features from different fields are first introduced and then heuristic rules are proposed to determine these three key scenarios.

5.2.1. Audio Features Selection

In order to achieve the accuracy of the scenario classification, audio information is an important and necessary clue. Because feature extraction is very important for audio content analysis, the extracted features should capture the temporal and spectral structure of different audio classes from each scenario talked above. Here, following features are selected to complete the task of audio clip classification, such as zero-crossing rate, energy envelope, spectrum flux, band periodicity, mel-frequency cepstral coefficients, spectral power, and linear prediction based cepstral coefficients.
Furthermore, in our experiment, all audio clips are divided into non-overlapping sub-clips. A sub-clip is of one second duration and is further divided into forty twenty-five millisecond-long frames, and short-time energy envelope entropy. The classification is performed based on these one-second sub-clips.

- **Zero-crossing Rate**. Zero-crossing rate is a simple measure of the audio signal frequency content. The N-length short-time zero-crossing rate of the n^{th} audio frame $s(n)$ is defined as:

$$ZCR(n) = \frac{1}{N} * \sum_{t=n-N+1}^{n} \frac{|\operatorname{sgn}\{s(t)\} - \operatorname{sgn}\{s(t-1)\}|}{2} w(n-t) \tag{12}$$

where $w(n)$ is a rectangular window, and

$$\operatorname{sgn}\left[s(n)\right] = \begin{cases} 1 & s(n) \geq 0 \\ -1 & s(n) < 0 \end{cases} \tag{13}$$

- **Energy Envelope.** Energy envelope is used to calculate the global temporal information. It can be computed in following way: third order Butterworth low-pass filtering of the analytical signal root mean square amplitude of each audio frame:

$$EE(n) = \sqrt{\frac{1}{N} \sum_{t=1}^{N} \left[s_t(n)\right]^2} \tag{14}$$

where N is the number of sample points in the n^{th} audio frame.

- **Spectrum Flux**. Spectrum flux is defined as the two-norm of the frame-to-frame spectrum amplitude difference vector:

$$SF(n) = \left\| \left|M_f(n)\right| - \left|M_f(n+1)\right| \right\| \tag{15}$$

where $|M_f(n)|$ is the magnitude of the FFT of the n^{th} frame at frequency value f. Both magnitude vectors are normalized in energy. Spectrum flux is a measure of spectral change between the adjacent two frames.

- **Band periodicity**. Band periodicity describes the property of each sub-band. In this chapter, we consider four sub-bands including 500~1000Hz, 1000~2000Hz, 2000~3000Hz, and 3000~4000Hz respectively. The periodicity property can be represented by the maximum local peak of the normalized correlation function.

- **Mel-Frequency Cepstral Coefficients (MFCCs)**. It has been proved that the mel-frequency cepstrum is a useful and highly effective feature in modeling the subjective pitch and frequency content of audio signals. The MFCCs are computed from fast Fourier transformation:

$$\begin{cases} MFCC(n) = \sqrt{\dfrac{2}{J}} \sum_{j=1}^{J} \left\{ \left(\log S_j\right) \times \cos\left[t\left(j-0.5\right)\dfrac{\pi}{J} \right] \right\} \\ t = 1, 2, ..., T \end{cases} \tag{16}$$

where the parameter J denotes the number of band-pass filters, and the parameter T means the order of the cepstrum. In our scheme, J and T are set to be 24 and 12 respectively, namely the 24 band-pass filters and 12-order MFCCs are used.

- **Spectral Power**. Spectral power of each audio frame is computed with a Hanning window $h(n)$:

$$h(n) = \sqrt{\frac{2}{3}} \times [1 - \cos(2\pi \frac{n}{N})] \tag{17}$$

The spectral power of the n^{th} audio frame $s(n)$ is:

$$SP(k) = 10\log_{10}\left[\frac{1}{N} \left\| \sum_{n=0}^{N-1} s(n)H(n)\exp(-j2\pi\frac{nk}{N}) \right\|^2 \right] \tag{18}$$

- **Linear prediction based cepstral coefficients (LPCCs)**. LPCCs are utilized to represent the timbre information of the voice components. The basic idea behind linear predictive analysis is that an audio frame can be approximated as a linear combination of past audio frames. By minimizing the sum of the squared differences over a finite interval between the actual audio frames and the linear predictive ones, a unique set of predictor coefficients can be determined.

5.2.2. Audio signal Classification

After removing silent clips, audio clips are classified into two categories firstly, i.e. speech and non-speech, based on the information of zero-crossing rate, energy envelope, and spectrum flux. Then, speech clips are classified into emotional voice and common voice based on the spectrum flux, band periodicity, mel-frequency cepstral coefficients, and non-speech

clips are classified into special sound and music based on the spectral power, and Linear prediction based cepstral coefficients. Finally, special sound is further classified into several classes, including gunshot, explosion, scream, beating, crashing of glass, and rubbing of tires using hidden markov models (HMM). Figure 11 illustrates the classification scheme.

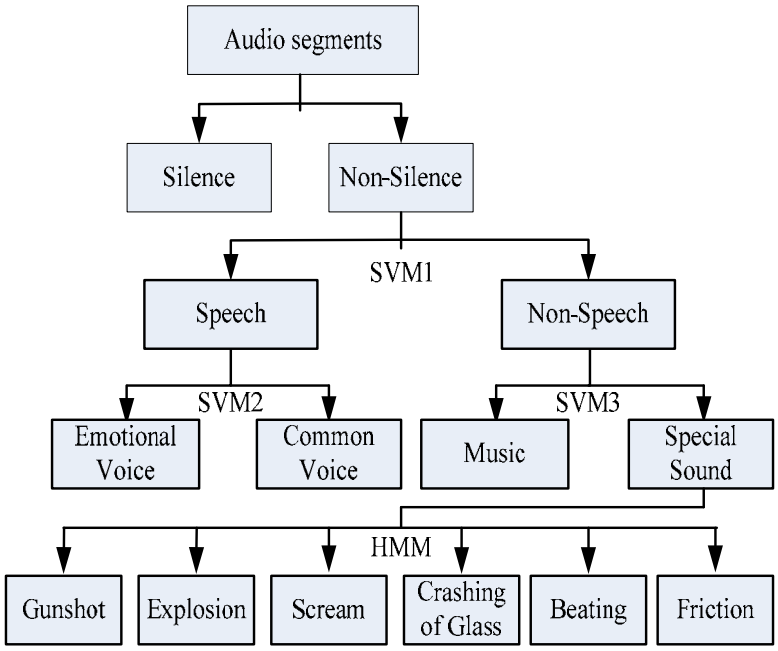


Fig. 11. Audio Classification scheme.

For support vector machines-based classification, features are first combined to construct a feature vector. Then, the mean and standard deviation of these feature vectors over all forty audio frames are calculated, and these statistics compose another feature vector. Finally, this new feature vector is normalized by its standard deviation of training data. The normalized feature vector is considered as the final representation of one-second audio signal.
The information of timbre and rhythm is utilized in the generative model for recognition, namely hidden markov models. Timbre allows one to tell the difference between sounds at the same loudness made by different objects. Each kind of timbre is denoted by one state of HMM, and represented with the Gaussian mixture density. Here, rhythm is adopted to represent the change pattern of timbres, and is denoted by transition and duration parameters in HMM.

5.2.3. Visual Features Selection

- **Face Information**. The occurrence of face is a salient feature in video, as it means the present of human in the scene. The size of a face is also a hint for the role of the person, i.e., a large face denotes that this person is in the center of attention. In the experiment, we use the face detection algorithm proposed by Li et al. [23] which performs reasonably good

for faces with different scales in the video. As a result of the face feature extraction process, we obtain the position and size of each detected face, and the number of hits.

- **Illumination Intensity**. Reference [24] indicates "the amount and distribution of light in relation to shadow and darkness and the relative tonal value of the scene is a primary visual means of setting mood." The amount of light in scene *Sc(k)*, here, is described as the average illumination intensity:

$$\begin{cases} II(k) = \dfrac{\sum_i ShAvgInt(i) \times ShLen(i)}{N(k)} \\ ShAvgInt(i) = \dfrac{\sum_j KfInt(j)}{N(i)} \end{cases} \tag{19}$$

where *ShAvgInt(i)* is the average illumination intensity of the entire key frames in the i^{th} shot, *ShLen(i)* is the length of the i^{th} shot in terms of frames, and *N(k)* is the total number of frames within the k^{th} scene. Furthermore, *KfInt(j)* is the illumination intensity of the j^{th} key frames in the i^{th} shot, and *N(i)* is the total number of frames within the i^{th} shot.

- **Activity Intensity**. The activity intensity indicates the tempo in video. For example, in conversational scenario, the activity intensity is relatively low. On the other hand, in action scenario, the activity intensity is relatively high. The activity intensity of the k^{th} scene is:

$$AI(k) = \frac{1}{N_k - 1} \sum_{l=1}^{N_{k-1}} \min \left[\sum_{i=0}^{i=L-1} \sum_{j=0}^{j=L-1} \left(H_{ij}^{l}, H_{ij}^{l+1} \right) \right] \tag{20}$$

- **Average Duration**. Similar to activity intensity, average duration in terms of frames is another measurement of the video tempo and is computed as follows:

$$AD(k) = \frac{1}{N_k} \sum_{l=1}^{N_k} ShLen(l) \tag{21}$$

5.2.4. Dialogue Scenario Determination

- **Dialogue Scenario Determination.** To locate the dialogue scenarios, we first adopt the information of face to detect shot sequence with alternately recurring visual contents, which include similar size, similar position and same number. Figure 12 shows an example of dialogue-like scenario.

Fig. 12. A dialogue-like scene detected using face information.

Given these dialogue-like scenes, we exploit their corresponding audio information to further make sure they are actual conversation contents. Specifically, we should differentiate speech signals from music and other sounds. Here, a simple classification method is utilized to complete the task of discrimination using zero-crossing rate, energy envelope, and spectrum flux as shown in Figure 11.

- **Emotional Dialogue Scenario Determination.** Among many dialogues, emotional conversations often attract viewers' attention and effect upon the flow of story. To discriminate the emotional contents from common ones, two acoustic features including average pitch frequency and temporal intensity variations are used. The first feature is estimated by 12-order linear prediction based cepstral coefficients, and the second one is represented by the variance of spectral power levels over all forty signal segments within one-second audio sub-clip.

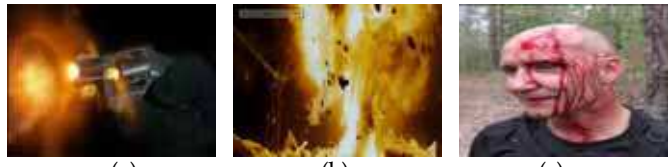
5.2.5. Active Scenario Discrimination

- **Active Scenario Discrimination.** Similar to dialogue scenario, active scenario is another conceptually meaningful story content. Among active scenarios, gunfight scenario, beating scenario, and chasing scenario are often the most interesting events and can instantly attract viewers' attention in films. Therefore, based on the recognition of active scenario by integrating audio-visual signatures, three specific and distinct events are identified one-by-one.

According to the characteristics of active scenario as talked above; these scenes with average duration less than twenty-five frames and average activity intensity are identified as active scenarios.

Among active scenarios, gunfight, beating and chasing events are three important types and mostly the climaxes in feature films. Next, we will detect these important active scenarios using their unique audio-visual signatures.

- **Gunfight Scenario Discrimination.** Gunfire, explosion, and bleeding are the most typical visual features of gunfight scenarios as shown in Figure 13.



(a) (b) (c)

Fig. 13. The most typical visual features of gunfight scenario including (a) gunfire; (b) explosion; (c) bleeding.

Compared to gunfire cases, flames from an explosion show longer duration and cover wider areas. However, flames from an explosion and gunfire both have dominant yellow, orange and/or red color histogram. Hence, a predefined color table containing a certain range of color values is adopted to identify the gunfight-like scenario.

Since some violent actions, such as beating, gunshot and explosion can result in bleeding, bleeding is considered as another violence-related visual feature of gunfight event. We detect bloody color pixels using simple pixel-matching with the predefined color table.

Since other events may have similar visual features as gunfire, explosion and bleeding, the audio information provides a supplement to the detection of gunfight scenario. A distinct feature of gunfight scenarios is the unique sound track. Specifically, given the audio track for successive gunfight-like shots, we discriminate its class based on a hidden markov model. The likelihood ratio between the input audio track and the defined sound classes is calculated to determine which class the associated sound belongs to as shown in Figure 11.

- **Beating or Chasing Scenario Identification.** In general, beating or chasing events are inherently accompanied by unique sound (e.g., beating, rubbing of tires, etc.). In particular, for active scenario, we identify its specific class (beating or chasing) based on the likelihood ratio between its audio track and the given sound classes as shown in Figure 11.

5.2.6. Suspense Scenario Detection

- **Suspense Scenario Detection**. Suspense scenarios are often the most events and instantly attract a viewer's attention in horror and detective genres. According to the unique characteristics of suspense scenario as talked about in the introduction of Section five, a scenario can be declared as a suspense scenario if following criterions are satisfied simultaneously:

(1). Average illumination intensity is less than 50;
(2). There exist shots with audio energy envelope change suddenly from 5 to over 50;
or / both
There exist shots with activity energy change instantly from 5 to over 100.

5.3. Scenario Transition Form

As aforementioned, there exist three forms of scenario transition: temporal transition, spatial transition and rhythmic transition. Next, we formulate each of them.

- Generally speaking, viewers can understand the temporal transition with respect to the different number of characters appearing in two scenarios *k(i)* and *k(j)* with respect to the change of number of face:

$$TT(k_i,k_j)=\left|\sum_{s=1}^{S}C(s,m,k_i)-\sum_{t=1}^{T}C(t,n,k_j)\right| \tag{22}$$

where $S(m,k_i)$ is the last shot of *k(i)* and $S(n,k_j)$ the first shot *k(j)*. Temporal transition between two scenarios is discriminated f the following inequality is true:

$$TT(k_i,k_j)>\frac{1}{S+T}\left[\sum_{s=1}^{S}C(s,m,k_i)+\sum_{t=1}^{T}C(t,n,k_j)\right] \tag{23}$$

where *S* and *T* are the number of key frames in $S(m,k_i)$ and $S(n,k_j)$, respectively.

- Spatial transition depicts the change of positions in successive scenarios for the same characters, which can be determined in terms of the change color information of background regions. The background regions are obtained by excluding the face region of each character, which is feasible due to most characters are shown in close-up views in a film. The intensity of spatial transition between two scenarios *k(i)* and *k(j)* is formulated as below:

$$\begin{aligned}ST(k_i,k_j)=&\left|\frac{1}{S'}\sum_{s=1}^{S'}RA(s)-\frac{1}{T'}\sum_{t=1}^{T'}RA(t)\right|+\left|\frac{1}{S'}\sum_{s=1}^{S'}GA(s)-\frac{1}{T'}\sum_{t=1}^{T'}GA(t)\right|\\&+\left|\frac{1}{S'}\sum_{s=1}^{S'}BA(s)-\frac{1}{T'}\sum_{t=1}^{T'}BA(t)\right|+\left|\frac{1}{S'}\sum_{s=1}^{S'}LA(s)-\frac{1}{T'}\sum_{t=1}^{T'}LA(t)\right|\end{aligned} \tag{24}$$

where *RA(s)*, *GA(s)*, *BA(s)*, and *LA(s)* are average red, green, blue, and luminance values in the background of the s^{th} *key* frame, respectively. S' and T' are the number of key frames in

S(m,ki) and *S(n,kj)* for the same character, respectively. There exists spatial transition between *k(i)* with the last shot *S(m,ki)* and *k(j)* with the first shot *S(n,kj)* when the following inequality holds:

$$ST(k_i,k_j) > \frac{1}{2}\left|\frac{1}{S'}\sum_{s=1}^{S'} RA(s) + \frac{1}{T'}\sum_{t=1}^{T'} RA(t)\right| + \frac{1}{2}\left|\frac{1}{S'}\sum_{s=1}^{S'} RA(s) + \frac{1}{T'}\sum_{t=1}^{T'} RA(t)\right| + \frac{1}{2}\left|\frac{1}{S'}\sum_{s=1}^{S'} GA(s) + \frac{1}{T'}\sum_{t=1}^{T'} GA(t)\right| + \frac{1}{2}\left|\frac{1}{S'}\sum_{s=1}^{S'} LA(s) + \frac{1}{T'}\sum_{t=1}^{T'} LA(t)\right| \quad (25)$$

- Rhythmic transition in terms of duration is adopted to represent the tense or clam atmosphere. The intensity of rhythmic transition between scenarios *k(i)* with the number of *M* shots and *k(j)* with the number of *N* shots is computed using following equation:

$$RT(k_i,k_j) = \left|\frac{1}{M}\sum_{m=1}^{M} S(m,ki) - \frac{1}{N}\sum_{n=1}^{N} S(n,kj)\right| \quad (26)$$

There can be declared as the rhythmic transition if the condition in following inequality is true

$$RT(k_i,k_j) > 2\left|\frac{1}{M}\sum_{m=1}^{M} S(m,ki) + \frac{1}{N}\sum_{n=1}^{N} S(n,kj)\right| \quad (27)$$

5.4. Scenario Transition Intensity

As aforementioned, the intensity of the flow between two scenarios to the whole story is formulated as the Intensity of Flow (IoF) function. At the same time, according to the discussion in Section 4, the flow of a story consists of various aspects, such as temporal transition, spatial transition, and rhythmic transition, and each metric is normalized between 0 and 1. Therefore, the form of IoF function between scenarios *k(i)* and *k(j)* is the weighted sum of these three metrics as expressed in Equation (16):

$$IoF(k_i,k_j) = \alpha * TT_n(k_i,k_j) + \beta * ST_n(k_i,k_j) + \gamma * RT_n(k_i,k_j) \quad (28)$$

where $TT_n(k_i,k_j)$, $ST_n(k_i,k_j)$, and $RT_n(k_i,k_j)$ are the corresponding normalization form of $TT(k_i,k_j)$, $ST(k_i,k_j)$, and $RT(k_i,k_j)$, respectively. $\alpha+\beta+\gamma=1$.

5.5. Video Skimming Approach

After calculating all pairs of transition intensity of a story-oriented video and given a user-defined target length, the next step for us is to choose the pairs of scenarios with the maximum total IoF value among all possible candidate scenarios.

Many approaches can be used to create video skimming based on the values of intensity of flow among detected key scenarios. We have developed a straightforward scenario-based approach to generate skimming, which does not require complex heuristic rules. Skimming segments of scenario around key frames are selected according to the given skimming duration. The process of skimming segment selection is illustrated in Figure 14.

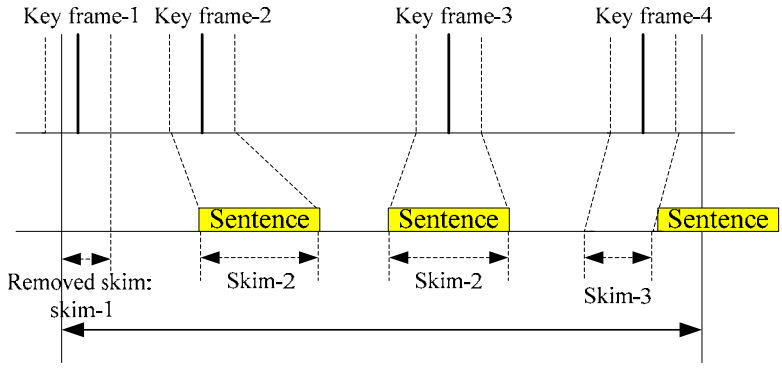


Fig. 14. Video skimming approach.

One crucial element in generating video skimming is the integrity of sentence. It must be uncomfortable for viewers to hear an interrupted sentence while watching a video skimming. Therefore, for the purpose of making a video skimming smoother, we should avoid splitting the speech within a sentence into several parts. In this case, it is indispensable to identify sentence boundary precisely for video skimming. In this chapter, sentence boundary identification comprises following steps.

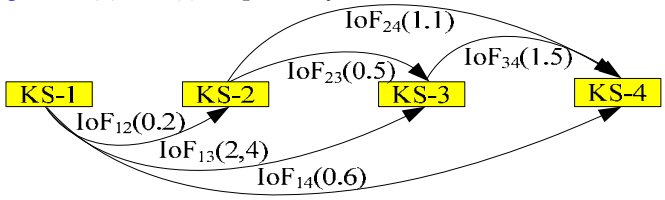
- Discriminating pause segments from non-pause ones using audio energy and zero-crossing rate.
- Smoothing results with respect to the minimum pause duration and the minimum speech duration.
- Determining sentence boundary by longer pause duration.

Besides IoF value, scenario boundary, sentence boundary and key-frames, only four simple rules are used to create video skimming, as shown in Figure 14. The process of video skimming should comply with several criteria described as follows.

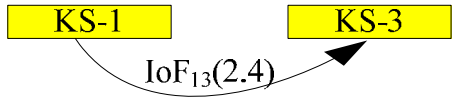
- Given a user-defined target length, pairs of key scenarios with the maximum total IoF value among all possible candidate scenarios are chosen to create video skimming.
- Any video skimming segment should not be less than one second due to the following two aspects. On the one hand, any segment no more than one second is too short to convey message. On the other hand, too short skimming segment may bring annoying impact on human understanding on video content.
- The length of each scenario skimming segment is proportional to the number of key frames in the scenario. Specifically, the default duration of any skimming segment centered on key frame is set to be one second.
- If a skimming segment is beyond the scenario boundary, it will be removed like skim-1 in Figure 14.
- The skimming segment boundaries should be adjusted appropriately in terms of the speech sentence boundaries to prevent from splitting a speech sentence into several parts. Here, the adjustment includes two aspects. One is aligning to the sentence's boundary like skim-2 in Figure 14; the other is evading the sentence's boundary like skim-3 in Figure 14.

Figure 15 shows examples of the story-oriented video skimming. Figure 15 (a) is a graph of the transition intensity among key scenarios of a story-oriented video. The durations of each

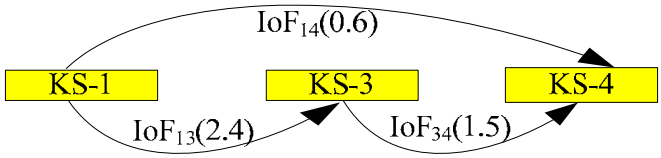
scenario skimming *ks1*, *ks2*, *ks3*, and *ks4* are 5 seconds, 10 seconds, 15 seconds, and 10 seconds, respectively. The skimming with the target durations of 7 seconds and 10 seconds are shown in Figure 15 (b) and (c), respectively.



(a): Intensity of flow of a video.



(b): A skimming with target length of 7 seconds.



(c): A skimming with target length of 10 seconds.

Fig. 15. Examples of the skimming of a story-oriented video.

6. Experimental Results

In this section, we will discuss some implementation issues in practical application and evaluate our approach on various genres of films with existing state-of-the-art methods.

6.1. Experimental Settings

Totally seven full-length movies in MPEG-1 format are selected to evaluate the performance of the proposed algorithm. Each video track is analyzed at 25 frames per second with a resolution of 320×240, while the sound track is processed at the sampling rate of 22 kHz with mono-channel, 16-bit precision. As shown in Table 2, the data set consists of various types of contents, which would demonstrate that the proposed algorithm can work on different kinds of movies. All the experiments are performed on an Intel Pentium IV 3.0 GHz machine and 1G memory computer running Windows XP.

No.	Name	Genre	Length (hh:mm:ss)
1	Mission Impossible III	Bodyguard	1:55:23
2	X Man III	Action	1:44:03
3	Walk in the Clouds	Family	1:42:14
4	The Girl Next Door	Love	1:48:10
5	The Ring	Horror	1:53:43
6	The Sound Of Music	Musical	2:54:33
7	N Death on the Nile	Detective	2:11:50

Table 2. Summary of the testing dataset.

6.2. Experiment I: Scenario Boundary Detection

The performance of scenario boundary detection (SBD) is important for video skimming because the proposed video skimming scheme is based on the scenarios. In order to evaluate the effectiveness of the proposed SBD approach, we compare it with the SBD approach proposed by Rasheed *et al.* [25] and Zhao *et al.* [26].

To get the ground truth of scenarios, ten graduate students are invited to watch the movies and then give their own scene boundaries. The ground truth used for the experiments is the intersection of their segmentation. Generally speaking, there is no such a clear boundary between two adjacent scenes in movies due to film editing effects. Therefore, the most commonly used criterion, Hanjalic's evaluation [19] is here used to match the ground truth with the detected ones: if the detected scene boundary is within four shots from the boundary detected manually, this boundary is counted as a correct boundary.

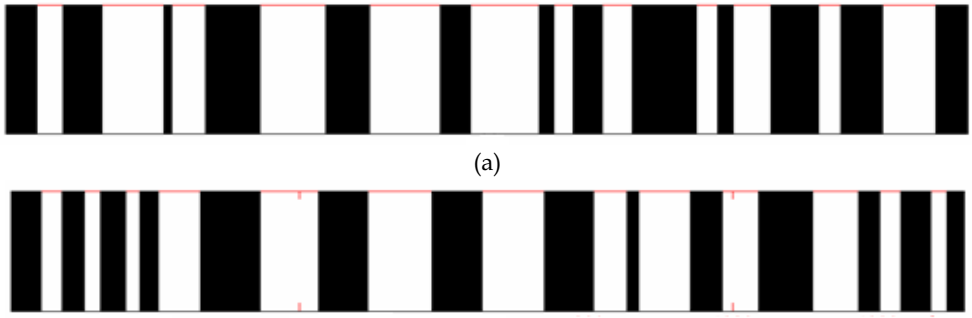
We use 'Precision', 'Recall' and 'F1-Score' to evaluate the performance of those techniques. The values of recall and precision are in the range of [0, 1]. The higher recalls indicate a higher capacity of detecting correct shots, while the higher precisions indicate a higher capacity of avoiding false matches.

No.	Proposed		[25]		[26]	
	Pre	Re	Pre	Re	Pre	Re
1	0.755	0.809	0.641	0.695	0.712	0.763
2	0.741	0.814	0.572	0.609	0.676	0.638
3	0.813	0.849	0.713	0.737	0.708	0.718
4	0.835	0.878	0.687	0.623	0.722	0.693
5	0.814	0.839	0.706	0.723	0.683	0.706
6	0.843	0.862	0.734	0.713	0.714	0.704
7	0.834	0.857	0.753	0.741	0.727	0.707

Table 3. Comparison of Scenario Boundary Detection (Pre: Precision, Re: Recall).

The results are given in Table 3. From Table 3, it can be seen that average F1-Score of the proposed approach, Backward Shot Coherence method and the Normalized Cuts method are 0.824, 0.688, and 0.704 respectively. That is, the proposed approach exhibits a gain 19.7% and 16.9% of the average F1-Score compared with the method in [25, 26] respectively. The reason why the proposed approach gains a large improvement in all the evaluating measures is based on following two phases. On the one hand, the temporal constraint is integrated into the spatial coherence clustering in the procedure of scene segmentation. On the

other hand, the process of forward and backward clustering also helps to improve the performance.



(a)

(b)

Fig. 16. Example of scenario detection. (a): Ground truth scenes and (b): detected scenes of the movie 'X Man III'.

Figure 16 shows the detail scene detection results of the movie 'X Man III'. The upper row indicates the ground truth scenes represented by alternating black / white stripes, and the bottom row is the detected results using the proposed scheme.

6.3. Experiment II: Key Scenario Identification

Since the proposed video skimming scheme is based on the flow of story between key scenarios, the detection precise of key scenarios have an important impact on final video skimming. Like the method used in providing the ground truth of scenario boundary, ten volunteers are invited to manually label key scenario.

Table 4 lists experimental results of the identification of key scenarios. As seen in Table 4, the average F1-Score is 0.855, which clearly demonstrate that the proposed scheme can detect and classify video scene into categories. That is to say, the selected features for describing scene content and the way of deciding scene type are satisfactory.

Conversation scene	Precision	0.897
	Recall	0.863
	F1	0.880
Suspense scene	Precision	0.863
	Recall	0.843
	F1	0.853
Action scene	Precision	0.846
	Recall	0.824
	F1	0.835

Table 4. Experimental results of Key Scenario Identification.

Figure 17 shows some examples of different types of scenario transitions.

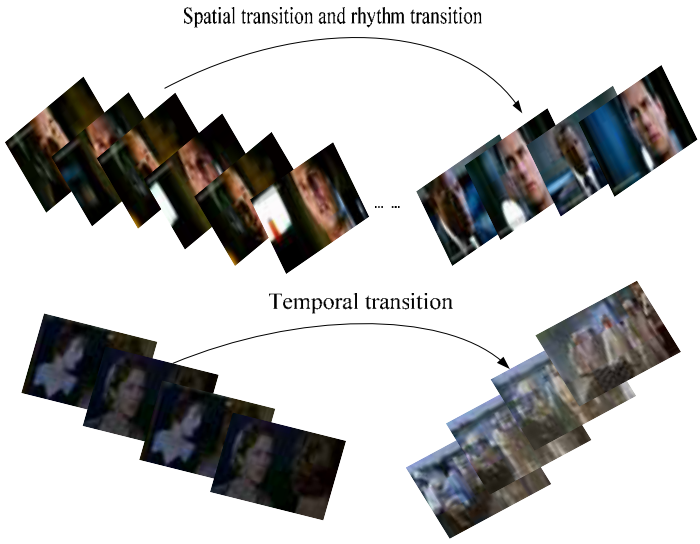


Fig. 17. Examples of scenario transition.

6.4. Experiment III: Video Skimming

Generally speaking, a good video skimming should be as short and as information as possible. However, it is difficult to achieve the two objectives at the same time. In this chapter, three criteria are adopted to evaluate the algorithm performance.

For a good video skimming, the first criterion is required to correctly select essential segments for viewers to grasp the clue of the flow of a story. We set the first criterion as informativeness including coverage and conciseness, where *coverage* means a skimming should include all the important segments of a story and *conciseness* means a skimming should comprise only the necessary segments. The second criterion is required to maintain the validity of interrelationships between segments. We set the second criterion as *coherence*, which denotes each segment of a skimming should be interrelated with others with respect to the whole story. Besides, *satisfaction* of the video skimming is also an important criterion. The last criterion appraises not only the smoothness of an image sequence, but also the integrality of speech sentence. Consequently, we will carry out two different experiments according to above discussed three criteria.

6.4.1. Verifying the First Criteria

This experiment aims to measure the precision and recall of each skimming compared to the manually produced ground truth by evaluating the coverage and conciseness of the proposed approach.

Table 5 shows the experimental results of precision and recall for each skimming with three different lengths (5, 10, 20 minutes). From this table, we can see that the overall average precision are 80.3%, 82.6%, and 84.2% for video skimming with the duration of 5, 10, and 20 minutes, respectively. The overall average recall are 80.8%, 82.9%, and 84.3% for video skimming with the duration of 5, 10, and 20 minutes, respectively. These numbers indicate that the proposed skimming scheme is effective in generating highly compact skimming.

No.	Precision			Recall		
	5 M.	10 M.	20 M.	5 M.	10 M.	20 M.
1	0.753	0.793	0.812	0.773	0.812	0.829
2	0.784	0.806	0.821	0.792	0.813	0.817
3	0.812	0.832	0.852	0.833	0.846	0.853
4	0.836	0.848	0.863	0.827	0.846	0.864
5	0.805	0.827	0.843	0.826	0.819	0.832
6	0.841	0.856	0.872	0.824	0.843	0.862
7	0.793	0.817	0.834	0.781	0.826	0.841
Avg.	0.803	0.826	0.842	0.808	0.829	0.843

Table 5. Results of informativeness. (M.: minute).

6.4.2. Evaluating the last two Criteria

We perform subject tests to measure the *coherence* and *satisfaction* by evaluating viewers' satisfaction and preference over skimming sequences with different duration. Twenty volunteered subjects, including 11 males and 9 females, are invited to participate in the experiment. They are required to specify their preferences to each skimming sequence according to a list of prepared questions, where the measurement of preference is categorized into three following levels: "bad", "neutral" and "good".

To achieve precise scores, the subjects should be kept innocent of the video content before they view the video skimming. The subjects view the video skimming sequences of 5, 10, and 20 minutes and the original video in turn, and they are required to give a score to the test sequence when he/she finishes viewing a skimming sequence. To be fair, the subjects are given the chance to modify the scores assigned to the skimming sequences any time during the review process, because they would understand the video content in more details with the passage of time. With the scores assigned by subjects, corresponding average scores are then calculated, which reflect the viewers' satisfactory degree to different video skimming sequences.

L	Q1			Q2			Q3		
	B	N	G	B	N	G	B	N	G
5	5	20	75	5	12	83	6	14	80
10	3	13	84	2	8	90	2	10	88
20	0	10	90	0	9	91	0	9	91

Table 6. Performance Evaluation of Video Skimming (L.: length, B: Bad, N: Neutral, G: Good).

The experimental results of the subject preference of video skimming are list in Table 6. From this table, we can see that the produced skimming sequences show good performance in terms of viewers' satisfaction and preference. The units of later fine columns are '%'. The prepared issues are list as follows.

- Q1: Do you think that scenario segments in the skimming are mutually coherent?
- Q2: Do you think that scenario segments in such duration are all appropriate?
- Q3: Do you think that the meaning of speech sentence in each scenario segment is unambiguous?

7. Conclusions

Automatic video skimming is a powerful tool for video browsing and retrieval. In this chapter, we present a new approach for story-oriented videos. The proposed framework automatically generates video skimming that help viewers grasp the main content within a given duration, which is achieved by exploring the clues about human understanding of a story according to general scenario writing rules and editorial techniques. After the process of the detection of scenario boundaries and the identification of key scenarios, a video skimming is produced by selecting the maximum total Intensity of Flow value among all possible candidate scenarios satisfying the given target duration.

Experiments reveal that our framework obtains good performance for all the criteria of a story-oriented video skimming: informativeness, coherence, and satisfaction. The overall average precision and recall are both over 80% compared to the manually generated ground-truth. Furthermore, the subjective tests show that viewers can achieve high satisfaction and preference over the skimming sequences. We believe that these experimental results indicate our scheme is a feasible solution for the management of video repository and online review service.

Acknowledgement

This work is supported by the National High-Tech Research and development Project of China (973) under Grant No. 2006CB303103, and also supported by the National High-Tech Research and development Project of China (863) under Grant No.2009AA01Z330 and the National Natural Science Foundation of China under Grant No. 60833009.

8. References

[1] Pfeiffer, S., Lienhart, R., Fischer, S., and Effelsberg, W. Abstracting digital movies automatically. Int. J. Visual Communication and Image Representation, 7(4) (1996) 345–353.

[2] Hanjalic, A., Kakes, G., Lagendijk, R. L., and Biemond, J. Indexing and retrieval of broadcast news programs using Dancers. SPIE J. Electronic Imaging, 10(4) (2001) 871–882.

[3] Kim, J., Hyun S., Kang K., Kim M., Kim J., and Kim H. Summarization of News Video and Its Description for Content-based Access. Int. J. of Imaging Systems and Technology, 13(5) (2003) 267-274.

[4] Lie. W., and Lai, C. News Video Summarization Based on Spatial and Motion Feature Analysis. (PCM 2004).

[5] Babaguchi, N., Kawai, Y., and Kitahashi, T. Event based indexing of broadcasted sports video by intermodal collaboration. IEEE Trans. Multimedia, 4(1) (2002) 68–75.

[6] Chen, C., Wang, J., and Wang, J. Efficient News Video Querying and Browsing Based on Distributed News Video Servers IEEE Trans. Multimedia, 8(2) (2006) 257-269.

[7] Yeung, M, and Yeo, B. Video visualization for compact presentation and fast browsing of pictorial content. IEEE Trans. Circuits System and Video Technology, 7(5) (1997) 771–785.

[8] Hanjalic, A., and Zhang, H. An integrated scheme for automated video abstraction based on unsupervised cluster validity analysis. IEEE Trans. Circuits System and Video Technology, 9(8) (1999) 1280–1289.
[9] Ma, Y., Lu, L., Zhang, H., and Li, M. A user attention model for video summarization. (ACM MM 2002).
[10] Lee, S., and Hayes, M. An application for interactive video abstraction. (ICASSP 2004).
[11] Ma, Y., Lu, L., Zhang, H. Video Snapshot: A Bird View of Video Sequence. (MMM 2005).
[12] Valdés, V., and Martínez, J. On Video Abstraction Systems' Architectures and Modeling. (SAMT 2008).
[13] Luo, H., Gao, Y., Xue, X., Peng, J., and Fan, J. Incorporating feature hierarchy and boosting to achieve more effective classifier training and concept-oriented video summarization and skimming. ACM Trans. Multimedia Computing, Communications and Applications, 4(1) (2008) 1-25.
[14] Sundaram, H., and Chang, S Computable scenes and structures in films. IEEE Trans. Multimedia, 4(4) (2002) 482–491.
[15] Bordwell, D., and Thompson, K. Film Art: An Introduction. Technology report, McGraw-Hill Companies, 1996.
[16] Abutableb, A. Automatic thresholding of gray-level pictures using two-dimensional entropies. J. Computer Vision, Graphics, Image Processing, 47(1) (1989) 22 – 32.
[17] Smoliar, S., and Zhang, H. Content-based video indexing and retrieval. IEEE Mage. Multimedia, 1(2) (1994) 62–72.
[18] Yeo, B., and Liu, B. Rapid scene analysis on compressed videos. IEEE Trans. Circuits and Systems for Video Technology, 5(6) (1995) 533–544.
[19] Hanjalic, A., Lagendijk, R., and Biemond, J. Automated high-level movie segmentation for advanced video-retrieval systems. IEEE Trans. Circuits System and Video Technology, 9(4) (1999): 580–588.
[20] Bouthemy, P., Garcia, C., Ronfard, R., Tziritas, G., Veneau, E., and Zugaj, D. Scene Segmentation and Image Feature Extraction for Video Indexing and Retrieval. (VIIS 1999).
[21] Arai, H. Fundamental Techniques for Scenario Writing. Da-Bo Munhwa, 1987.
[22] Bordwell, D., and Thompson, K. Film Art: An Introduction. McGraw-Hill Companies. 1996.
[23] Chaisorn, L., Chua, T., and Lee, C. The segmentation of news video into story units. (ICME 2002).
[24] Aner, A., and Kender, J. Video summaries through mosaic-based shot and scene clustering. (ECCV, 2002).
[25] Rasheed, Z., and Shah, M. Scene detection in Hollywood movies and TV shows. (CVPR 2003).
[26] Zhao, Y., Wang, T., Wang, P., Hu, W., Du, Y., Zhang, Y., and Xu, G. Scene Segmentation and Categorization Using NCuts. (CVPR 2007).

Multimedia

Edited by Kazuki Nishi

ISBN 978-953-7619-87-9
Hard cover, 452 pages
Publisher InTech
Published online 01, February, 2010
Published in print edition February, 2010

Multimedia technology will play a dominant role during the 21st century and beyond, continuously changing the world. It has been embedded in every electronic system: PC, TV, audio, mobile phone, internet application, medical electronics, traffic control, building management, financial trading, plant monitoring and other various man-machine interfaces. It improves the user satisfaction and the operational safety. It can be said that no electronic systems will be possible without multimedia technology. The aim of the book is to present the state-of-the-art research, development, and implementations of multimedia systems, technologies, and applications. All chapters represent contributions from the top researchers in this field and will serve as a valuable tool for professionals in this interdisciplinary field.

INTECH
open science | open minds

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821